



Church, G., & Cliff, D. (2019). A Simulator for Studying Automated Block Trading on a Coupled Dark/Lit Financial Exchange with Reputation Tracking. In M. Affenzeller, A. Bruzzone, F. Longo, & G. Pereira (Eds.), *Proceedings of the 31st European Modelling and Simulation Symposium (EMSS2019)* (pp. 284-293). DIME University of Genoa.

Peer reviewed version

[Link to publication record in Explore Bristol Research](#)
PDF-document

University of Bristol - Explore Bristol Research

General rights

This document is made available in accordance with publisher policies. Please cite only the published version using the reference above. Full terms of use are available:
<http://www.bristol.ac.uk/red/research-policy/pure/user-guides/ebr-terms/>

A SIMULATOR FOR STUDYING AUTOMATED BLOCK TRADING ON A COUPLED DARK/LIT FINANCIAL EXCHANGE WITH REPUTATION TRACKING

George Church^(a) & Dave Cliff^(b)

Department of Computer Science
University of Bristol, Bristol BS8 1UB, UK.

^(a)gc14768@my.bristol.ac.uk, ^(b)csdtc@bristol.ac.uk

ABSTRACT

We describe a novel simulation of a contemporary real-world financial exchange: *London Stock Exchange (LSE) Turquoise*, and we also introduce a newly-created adaptive automated trading strategy called *ISHV*, which exhibits realistic behavior in situations where large orders can radically shift prices before transactions occur. LSE Turquoise is a recently-introduced platform where buying and selling takes place on a pair of coupled trading pools: a *lit* pool that is visible to all traders; and a *dark* pool where large "block" orders are hidden from sight until they are automatically matched with a counterparty, after which the transaction is then revealed. Orders from traders are routed to the lit or dark pool depending on their size, and on the reputation of the trader issuing the order. Unlike all other public-domain adaptive trading strategies, ISHV can alter the prices it quotes in anticipation of adverse price changes that are likely to occur when orders for block-trades are publicly visible: so-called *market impact*. LSE Turquoise is intended to reduce the negative effects of market impact; something that we test with our simulator. We extend the existing *BSE* open-source exchange simulator to incorporate coupled lit and dark pools, naming the new system BSELD. We show ISHV exhibiting market impact in a lit-only pool, and discuss how a Turquoise-style coupled dark pool reduces or eliminates that impact. We also show results from a Turquoise-style reputation-tracking mechanism, which can be used for modulating trader access control to the dark pool.

Keywords: financial markets, automated trading, dark pools, economic simulation.

1. INTRODUCTION

In contemporary global financial markets, traders routinely interact by remotely accessing an electronic trading venue, an online financial exchange, where they can post various types of order that indicate the prices at which they seek to sell or to buy some quantity of a particular tradable asset. They do this in the hope of finding one or more counterparties, other traders from whom they can buy, or to whom they can sell; interaction mediated via the trading venue allows potential counterparties to negotiate and agree a fair price for each transaction. In this minimally simple characterization, electronic exchanges are nothing more than digital implementations of the open-outcry trading pits that most

major financial exchanges operated before the advent of computerized trading. And, in turn, those trading pits were strikingly similar to the impromptu haggling between gatherings of buyers and sellers that has been an everyday occurrence at public markets and exchanges and souks and bazaars for many centuries.

However, once major financial exchanges had switched from physical trading floors to virtual ones, opportunities then opened up for novel structures of market, for bringing buyers and sellers together to identify counterparties and for them to jointly agree a fair price for their transactions, in ways that are different from, or would simply not have been possible, if the exchange had not virtualized. That is, electronic markets can sometimes be created to offer trading-venue functionality that would have been difficult or impossible to achieve in a traditional bricks-and-mortar exchange.

This is already trivially the case in major financial-trading centers such as New York, London, Tokyo, and Hong Kong, where a few decades ago there would be only one major national exchange for trading equities (stocks and shares) and yet where today a trader typically has a choice of several venues on which any one trade could be executed (and, in some countries, where the venues are required by law to pass a trader's order to competitor-venues if that gives the trader a better price for her deal). Many of the new alternative trading venues have been created post-virtualization, after the economic and regulatory barriers to entry were lowered by making it feasible for trading venues to be operated from out-of-town data-centers rather than prime real-estate in the center of major-city financial districts. However, such developments only scratch the surface of the new opportunities offered by fully electronic trading venues.

Less trivially, in recent years innovative operators have offered new forms of online exchanges that are designed specifically to counteract aspects of traditional markets that had previously made business difficult for traders. Traders working for investment banks or major fund-management companies, where their jobs require them to routinely handle "block orders", i.e. individual orders for very large quantities of an asset, face a particular problem known as *market impact*. A single order to buy 100,000 shares in IBM will often not affect the price of IBM stock in the same way that 100 bids for 1,000 shares of IBM do. Major venue-operators such as the London Stock Exchange (LSE) now operate electronic markets that are

custom-designed to reduce or eliminate the negative effects of market impact for block traders.

In brief, market impact is the name given to the effect experienced by traders when they try to execute a block (i.e., bulk) order, an order with a quantity so large that the sudden increase in supply (for sell orders, i.e. *asks*) or in demand (for buy orders, i.e. *bids*) means that other traders in the market — potential counterparties to the trade — recognize that the order is so big that it will shift the price of the asset in question: those potential counterparties then (entirely rationally) adjust the prices that they are willing to trade at, in a direction that is less attractive to the trader trying to initiate the block trade. This can be thought of as the counterparties guessing or estimating what price the asset will shift to and trade at *after* the block transaction has executed, and then setting that price for the transaction with the block trader *before* the price has actually shifted.

An example will help to illustrate this. Say that the current price for a tradeable asset with ticker-symbol WXYZ is roughly \$10, and that the usual size of a deal, the usual quantity traded in WXYZ, is 100 units. Then say that our trader Anne is looking to sell 20,000 units — 200 times bigger than a regular-sized order: this definitely qualifies Anne as a block trader, for this particular deal. Before she does anything, Anne checks her trading screen and sees buy orders on the exchange from Bob, Carrie, and Dee: they are bidding at \$9.98, \$9.97, and \$9.95, respectively. Anne asks these three potential buyers for their price on 20,000 WXYZ and they immediately respond with lowered prices of \$7.03, \$6.95, and \$7.02, respectively. This means that Anne has seen a 30% reduction in the price she could get for her sale, merely because she revealed the size of her intended trade: the size caused her potential counterparties to factor into the deal their expectations of how the price would fall as a consequence of the sudden major increase in supply of WXYZ that would result from Anne's deal executing. This 30% fall when selling is the *market impact*, the price moving against Anne. If Anne had instead been buying a similarly bulk deal, she would very likely have suffered impact in the other direction: the quoted price would have moved up, as the potential counterparties priced-in the effects of increased demand. If instead Anne had a way to signal her desire to trade but without fully revealing the size of her deal, and had waited for a while, maybe she could have found a willing counterparty, another block-trader looking to buy a big block who was happy with Anne's asking price.

That issue, of matching big block-traders (to their mutual advantage) without giving away details of the full size of each block-trader's intended deal is something that has been addressed by exchange and trading-venue operators who offer what are known as *dark pool* exchange facilities. Dark pools offer access to financial liquidity, 'darkened' so that key details of trader's orders are concealed from the pool's participants and observers, with only the trading-venue operator knowing the full details of the position of each trader in the pool. The venue operator matches bids and offers in the pool against those from willing counterparties, notifies the buyer and the seller that they have matched, and details of the trade are then made public after it has executed.

Market impact is avoided because no-one can see in what quantity a participant is seeking to trade, until the trade is agreed and goes through. In contrast, traditional trading venues, exchanges with full transparency on quantities, are now known as *lit* trading pools.

In the past 15 years much of the trading activity on electronic markets has become highly automated. That is, many or all of the entities trading on the exchange are no longer human traders, but instead are autonomous algorithmic trading systems buying and selling with little or no direct human control. Automated trading systems running on electronic markets typically process such vast quantities of data, and have such very short reaction times, that no human could ever hope to match their performance. Financial institutions that operate profitable trading desks treat details of their automated algorithmic trading systems as tightly-guarded commercial secrets, for the obvious reason that this is how they make their money, their competitive advantage. Nevertheless, there is a body of published work, a series of peer-reviewed academic papers and reputable university-level textbooks, that gives some insight on how adaptive automated trading systems are constructed. Since the mid-1990s researchers in universities and in the research labs of major corporations such as IBM and Hewlett-Packard have published details of various forms of trading strategy, often incorporating machine learning methods so that the automated trader can adapt its behaviors to prevailing market conditions. Notable trading strategies in this body of literature include the MGD and GDX automated traders developed by IBM researchers (Tesauro & Das, 2001; Tesauro & Bredin, 2002); the ZIP strategy developed at Hewlett-Packard (Cliff, 1997); HBL (Gjerstad, 2003); AA (Vytelingum, 2006; Vytelingum *et al.*, 2008); Roth-Erev (e.g. Pentapalli, 2008); and ASAD (Stotter *et al.* 2013). However, for reasons discussed at length in a recent review of key papers in the field (Cliff, 2019) this sequence of publications concentrated on the issue of developing trading strategies for regular-sized orders: none of the key papers explored in (Cliff, 2019) look at trading strategies for outsize block orders, and none of them directly explore the issue of how an automated trader can best deal with, or avoid, market impact.

The lack of prior studies of automated trading strategies for block orders may be due to the complexity of the test-environment that is required to evaluate such strategies. An automated block trader needs to act and interact in a market environment in which there are other traders, potential counterparties, that are capable of making reasonable guesses or estimates for where the price of the tradeable asset will shift to after the block trade has gone through: that is, the other traders need to react to block orders by exhibiting behavior that is reasonably describable as market impact in response to the arrival of a block order, and none of the public-domain algorithms listed above does that. Second, the trading venue in which the various traders are interacting needs to support the processing of orders that vary over a significant range of sizes, so that some orders are "small" and others "large" — large enough to have an impact effect.

We report here on automated-trading experiments that are novel on two fronts, and which directly address that

complexity. First, we show how an existing trading strategy can be modified in such a way that impact effects are exhibited when a block order is submitted to a market populated by those traders. For our market environment, we use the free public-domain open-source BSE exchange simulator (BSE, 2012) which provides much of the functionality of a traditional trading venue and comes with pre-coded versions of a variety of automated trading strategies. We then report on our work altering and extending BSE so that it is a model of the *Turquoise Plato* trading venue recently announced by the London Stock Exchange (LSE). Turquoise is a trading platform founded in 2008 and is now majority owned by the London Stock Exchange Group (LSEG). Turquoise features a lit order book called Turquoise Lit™ and a coupled dark order book called Turquoise Plato™. Turquoise Plato incorporates a novel reputation-tracking mechanism to encourage desirable behaviors in the traders using the dark pool. We refer to the extended BSE as *BSELD*: *BSE* with *Lit* and *Dark* venues. The experiments we report here are the first to demonstrate that populations of automated traders can exhibit market impact effects in a traditional venue like BSE, and that the impact effect is reduced when the BSELD venue structure is used instead. Our simulator allows us to explore the effects that different aspects of the BSELD architecture have on the overall market dynamics, and to evaluate trading algorithms designed to operate in such a contemporary trading environment.

In Section 2 we review the motivations for our work and relevant previous literature. In Section 3 we demonstrate the way in which automated traders can alter their quote-prices in response to the arrival of individual block orders, which is a useful model of large-block market impact, and we show results that illustrate the impact effects in trading sessions on the traditional lit-only BSE. In Section 4 we compare and contrast the outcomes for block traders in the paired lit/dark market structure offered in BSELD: via a simple argument, it can be seen that the addition of an appropriately coupled dark pool will indeed greatly reduce (or eliminate) size impact effects. Section 5 then describes the reputation system.

2. BACKGROUND: FINANCIAL MARKETS

In contemporary global financial markets, traders routinely interact by remotely accessing an electronic trading venue, an online financial exchange. Financial exchanges typically allow traders to buy or sell varying quantities of any of a large number of tradeable assets, and different exchanges specialize in different asset-classes: for example, in London the primary market for equities (stocks and shares) is the London Stock Exchange; the primary market for metals is the London Metal Exchange; and the primary market for derivatives (tradeable options and futures contracts) is the ICE Futures Europe Exchange (originally founded as the London International Financial Futures Exchange, LIFFE, with its name subsequently changing as a result of a succession of corporate acquisitions and mergers). Exchanges typically operate as a venue in which buyers and sellers can interact, typically via buyers submitting *bids* and sellers submitting *asks* (also known as *offers*) with the exchange accepting the buyers' bid orders and

the sellers' ask orders, and implementing some kind of matching process that enables traders to identify a willing counterparty for a transaction that then takes place at an agreed price, one which both parties consider fair.

Without loss of generality, we will consider here the market for only a single, anonymous, tradeable asset, and we will concentrate on the data-structure at the heart of operating an exchange in almost any class of tradable asset: that data-structure is the *Limit Order Book* (LOB). The LOB is typically published by the exchange and distributed to all traders simultaneously any time it changes. At any one time the LOB shows a summary of unexecuted orders that have been submitted by traders active in that market. The orders summarized on the LOB are of a specific type, known technically as *limit orders*. A trader in a market can typically choose to submit an order to the exchange either to buy or sell at whatever the current best available price is, this order-type is known as a *market order*; or a trader can choose to submit a *limit order*, one that specifies a *limit price*, a price that the asset is not yet trading at. When a trader submits a limit order to the exchange, that order is then held on the exchange's books until a counterparty can be identified that is willing to trade at a price that is equal or better than the order's indicated limit price: the book on which the limit orders are recorded at the exchange is the LOB. Because orders are fundamentally either bids or asks, the LOB is routinely spoken about as being divided into two sides: the bid side and the ask side. The best bid on the LOB is the one with the highest limit price, and the best ask is the one with the lowest limit price: the LOB is usually shown with the best bid and ask at the top, and with less competitively priced orders arranged below them in numeric order of price (descending for bids, ascending for asks). The quantity of the asset available at each price is also indicated on each side of the LOB, although the identities of the traders behind each order are typically not shown, so the LOB acts as an aggregator and an anonymizer of individual orders.

The difference between the best bid price P_b and the best ask price P_a is known as the *spread*, and the mid-point of the spread (i.e., the arithmetic mean, $(P_a + P_b)/2$) is the *midprice*, which is very commonly used in situations where a single price value is required to summarize the state of the market for that asset. An illustration of a LOB is provided in Figure 1. Prices are usually quantized to some atomic resolution known as the tick-size: in very many exchanges the tick size is one cent or one penny, i.e. $1/100^{\text{th}}$ of a unit of fiat currency. For further details of exchanges and the LOB see e.g. Harris (2002) or Cartea *et al.* (2015),

When a trader submits an order to the exchange, if it is a bid with a price greater than the current best ask-price, or an ask with a price lower than the current best bid-price, then the order is said to *cross the spread*, and that is interpreted as an indication that the trader submitting the order wishes to trade at the best price on the counterparty side of the LOB – known as *lifting the ask* (for an incoming bid that crosses the spread) or *hitting the bid* (for an incoming ask that crosses the spread). That is, a spread-crossing limit order is effectively a market order. Because any spread-crossing quote issued by a trader is

immediately executed (i.e., immediately results in a transaction occurring, consuming some quantity of the limit orders on the counterparty side of the LOB), the best bid and best ask on the LOB at any one time will always be different values and hence values for the spread are bounded from below by the tick-size.

	BID	ASK	
Qty	Price	Price	Qty
1	100	200	1
1	99	201	1
1	98	202	1
1	97	203	1
1	96	204	1
1	95	205	1
1	94	206	1

Figure 1: a Limit Order Book (LOB) presented in the conventional format: the left-hand side of the LOB is the Bid-book, the right hand side is the Ask-book. Prices are displayed as integer numbers of cents, so the *tick-size* is \$0.01. On each side of the LOB, prices are arranged top-to-bottom in order best to worst, so the highest bid price and the lowest ask price are at the top of the book. The columns at outside left and right, labelled "Qty", show the quantity of items available at that price. In this example the *spread* is \$2.00-\$1.00=\$1.00, and the *midprice* is $(\$1.00+\$2.00)/2 = \$1.50$.

Over the course of the 20th Century, trading activity on major exchanges transitioned from open-outcry trading pits to remote, electronically mediated communications. In the trading pits, which had often operated for well over 100 years, human traders gathered in close physical proximity and shouted and gestured at one another to directly signal their bids and offers. As relevant technology became cheap and reliable, traders no longer gathered at the exchange but instead communicated remotely: first by telephone; then via dedicated computer networks; and latterly using Internet and Web channels. Nevertheless, at the turn of the 21st Century, almost all traders active in any one major financial market would be human. In the past 15 years, that situation has changed significantly and now most trades in "spot" (immediate-execution) financial markets for a wide range of assets are dominated by *robot traders*, automated software systems using artificial intelligence (AI) and machine learning (ML) techniques that have replaced the human traders because trading machines are much faster, much more reliable, and much cheaper than human employees. This development, the rise of automated trading (and its potential ill effects) has been documented and discussed by various authors including Arnuk & Saluzzi (2012), Bodek & Dolgoplov (2015), Narang (2013), Patterson (2013), and Rodgers (2016).

For the purposes of this discussion, a key initial academic paper was the surprising set of results published in the prestigious *Journal of Political Economy* by Gode & Sunder (1993): this introduced a minimally simple automated trading algorithm now commonly referred to as *ZIC*. Four years later two closely related research papers were published independently and at roughly the same time, each written without knowledge of the other: the first was by Cliff (1997) describing the adaptive trading-agent strategy known as the *ZIP* algorithm; the

second by Gjerstad & Dickhaut (1998), describing an adaptive trading algorithm now widely known as *GD*.

Gjerstad later moved to IBM's TJ Watson Labs where he helped set up a laboratory facility that was used in a study that generated world-wide media coverage when the results were published by Das *et al.* at IJCAI-2001, the prestigious *International Joint Conference on Artificial Intelligence*. This paper presented results from experiments exploring the behavior of human traders interacting with GD and ZIP robot traders, in a LOB-based financial exchange, and demonstrated that both GD and ZIP reliably outperformed human traders. Neither GD nor ZIP had been designed to work with the LOB, so the IBM team modified both strategies for their study. A follow-on 2001 paper by Tesauro & Das (two co-authors of the IBM IJCAI paper) described a more extensively *Modified GD* (MGD) strategy, and later Tesauro & Bredin (2002) described the *GD eXtended* (GDX) strategy. Both MGD and GDX were each claimed to be the strongest-known public-domain trading strategies at the times of their publication.

Subsequently, Vytelingum's 2006 thesis introduced the *Adaptive Aggressive* (AA) strategy which, in a paper published in the *Artificial Intelligence* journal (Vytelingum *et al.*, 2007), and in later AI conference papers (De Luca & Cliff 2012a, 2012b), was argued to be dominant over ZIP, GDX, and human traders, and hence the strongest-known public-domain robot trader. Recent work (Vach, 2015; Cliff, 2019) has called into question the supposed dominance of AA over GDX and ZIP, but all three of these trading algorithms can outperform human traders.

3. MODELLING MARKET IMPACT

The essence of market impact was sketched in the example story involving Anne, Bob, Carrie, and Dee that was given in Section 1. For the purposes of the discussion in this paper, here we will more precisely define *market impact* as the situation in which the market mechanism (e.g., the LOB) for some tradeable asset *A* shows the best price $B(t)$ available from potential counterparties at time *t*, and where at time t_1 a trader *T* signals an intent to transact a large quantity *Q* of asset *A* at a price equal to or better than $B(t_1)$, and this immediately results in $B(t_1+\Delta t)$ worsening with respect to the transaction intended by *T*, *before any actual transaction has yet taken place*. Here "better than $B(t)$ " means a *higher* price if *T* is buying, and a *lower* price if *T* is selling; and "worsening" means the price *B* *increasing* if *T* is intending to buy, and *decreasing* if *T* is intending to sell. Similarly, "a large quantity *Q*" can be interpreted as any quantity sufficiently large, relative to the quantity previously available at the then-current best counterparty price, to cause a significant shift in *B*: a factor of 10 or more is usually sufficient.

From this definition, it is clear that the quantities bid or offered at various prices in the auction are the driving factors of market impact. However, to the best of our knowledge, none of the automated trading strategies mentioned in Section 2 incorporate any reasoning about order-quantity effects: neither in the original publications that introduce each strategy, nor in any later publications

that describe subsequent use of those strategies. Because of this, one novel aspect of our work described here is our demonstration of size impact effects in markets populated by artificial trading agents: we are not aware of any previous publications that have demonstrated this.

In this paper we focus initially on exploring impact effects in markets populated by traders running an adapted form of the SHVR strategy that is built-in to BSE (see BSE 2012, Cliff 2018a). SHVR operates by attempting to always beat the best bid or offer: if a SHVR is selling, it will issue a quote that shaves one cent off the best ask price on the LOB so long as it is able to do so; and if SHVR is buying it will issue a quote adding one cent to the LOB's best bid price (thereby shaving one cent off its potential profit). SHVR is an attractive first choice because of its simplicity; extending the methods and results presented here to working with other more sophisticated adaptive strategies such as ZIP, GDX and AA are obvious topics for further work.

We extended the open-source code for SHVR traders such that their quote prices will, in appropriate circumstances, be determined by a function, an algorithm, that takes order-quantity into account. Our alteration involves making the traders sensitive to the market's current *microprice*. The microprice (see e.g. Cartea *et al.* 2015) is a statistic that is related to the midprice but whereas the midprice is calculated only from price data, the microprice is a quantity-weighted calculation. If the LOB's current best bid price and quantity are denoted by P_b and Q_b respectively, and the current best ask price and quantity by P_a and Q_a respectively, then the microprice P_μ is given by:

$$P_\mu = P_a(Q_b / (Q_a + Q_b)) + P_b(Q_a / (Q_a + Q_b))$$

When $Q_a = Q_b$ this reduces to the equation for the midprice P_m , but as the difference or *imbalance* between Q_a and Q_b grows so the difference between P_μ and P_m increases, with P_m remaining constant but P_μ moving in the direction that subsequent transaction prices can reasonably expect to head in, given the current imbalance in supply and demand and assuming all other things remain equal. This captures the intuitive notion that, in times of excess demand, competition among buyers is likely to push prices up toward the best ask price, while in times of excess supply, competition among sellers is likely to push prices down toward the best bid price: in each case, the microprice moves in the corresponding direction. An important factor to note here is that the microprice gives an indication of the expected direction of change in transaction prices, *before* any transactions affected by market impact have actually taken place. Figure 2 illustrates the effect that size imbalances at the top of the LOB can have on the microprice: The left-hand table illustrates the case where there is excess demand; the right-hand table shows a situation with excess supply. The midprice is the same in each case, but the size imbalance skews the microprice either down or up, depending on whether there is an excess of supply or demand. The Δ value indicated above each table is the difference between the midprice and the microprice.

To distinguish the original SHVR from the adapted/extended version, we will refer to the latter as

ISHV (pronounced *eye-shave*; the *I* can stand for *imbalance*, or for *impact*).

midprice= 150.0				midprice= 150.0			
microprice= 190.9				microprice= 109.1			
$\Delta = 40.9$				$\Delta = 40.9$			
		BID	ASK			BID	ASK
Qty	Price	Price	Qty	Qty	Price	Price	Qty
10	100	200	1	1	100	200	10
1	99	201	1	1	99	201	1
1	98	202	1	1	98	202	1
1	97	203	1	1	97	203	1
1	96	204	1	1	96	204	1
1	95	205	1	1	95	205	1
1	94	206	1	1	94	206	1

Figure 2: effects of size imbalances on the microprice.

In the current version of ISHV, the amount that the trader shaves off the current best counterparty price on the LOB (denoted by Δ_s , and which in the original SHVR was a constant, equal to the exchange's tick-size Δp) is determined by the imbalance at the top of the LOB, the difference between the midprice and the microprice, which we refer to as Δm :

$$\Delta m = P_\mu - P_m$$

Intuitively, if there is no imbalance, $\Delta m = 0$; if $\Delta m > 0$, the imbalance indicates that subsequent transaction prices will increase; and if $\Delta m < 0$, the indication is that prices will subsequently fall. This can be formalized in ISHV by defining an *impact function*, F , that maps from Δm to Δ_s , i.e. $\Delta_s = F(\Delta m)$. We have explored a number of forms for the impact function, and the results that we present here have been generated from a simple linear form expressed as pseudocode in Figure 3.

The rationale for this impact function is straightforward: at the macro level buyers and sellers react in symmetrically complementary ways; if the imbalance indicates that the price will be worsening for a trader, the trader reacts by attempting to shave a proportionately larger amount off of the best price on the LOB, thereby lowering its potential profit on the trade, but at least trying to "stay in the game", maintaining its chances of actually getting a deal. Conversely, if the imbalance indicates that prices will be improving for the trader (i.e., prices are likely to move in favor of the trader, because of excess quantity on the counterparty side of the book), the ISHV trader reduces the amount it shaves off the best price: here, it sets the shave-amount to the smallest possible, the exchange's tick-size Δp . Finally, for that reduction in shaving when the price is moving in the trader's favor to be meaningful, ISHV's "default" shave-amount at $\Delta m = 0$ (which was uniformly Δp for the original SHVR) the constant C should be > 1 : here we use $C=2$. In the current impact function, for both buyers and sellers, the scaling of Δ_s as $|\Delta m|$ increases is linear with $M=1$: avenues for further research include studying the effects of varying the values of C and M , and/or using a nonlinear form for $F(\Delta m)$.

To explore the extent to which a BSE market populated with ISHV traders can act as a reasonable model of

market impact, we devised experiments that are somewhat artificial in comparison to the dynamics of real markets, but in which the artificial constraints have been introduced to help clarify any market impact effects. We create a market with N buyers and N sellers, and initially all buyers and sellers are given assignments to trade in maximum quantities of one (so, after traders have issued quotes to the exchange, the quantity columns in the LOB resemble those of Figure 1), and with prices according to a fixed supply and demand schedule with a known underlying theoretical equilibrium price, denoted by P_0 . Traders are given a period of time to interact via the exchange's LOB-based auction mechanism, settling to a steady state, and then at a specific time a lone trader is given an assignment to either buy or sell a large quantity of the asset at a price which places that trader's order at the top of the LOB (creating a situation on the LOB much like those illustrated in Figure 2). This single large order can be thought of as a step-change "shock" to the market; albeit a quantity-shock rather than a price-shock. We then monitor what happens to the prices quoted on the LOB, and to any subsequent transaction prices, as the traders in the market respond and adapt to the post-shock conditions.

```

if (buying)
then if  $\Delta m < 0$ 
    then  $\Delta s = \Delta p$ 
    else  $\Delta s = C\Delta p + M\Delta m\Delta p$ 
    if  $P_b + \Delta s \leq p_{lim}$ 
    then  $q = P_b + \Delta s$ 
    else  $q = p_{lim}$ 

if (selling)
then if  $\Delta m > 0$ 
    then  $\Delta s = \Delta p$ 
    else  $\Delta s = C\Delta p - M\Delta m\Delta p$ 
    if  $P_a - \Delta s \geq p_{lim}$ 
    then  $q = P_a - \Delta s$ 
    else  $q = p_{lim}$ 

```

Figure 3: pseudocode for the ISHV imbalance-sensitive decision tree that sets Δs , the amount an ISHV attempts to shave off the current best price on the LOB, and then uses that to set q , the price that the trader quotes in the market. Δm is the difference between the microprice and the midprice; Δp is the exchange's tick-size; p_{lim} is the limit price on the current assignment (i.e., the customer order being worked); P_a is the price of the best ask on the LOB; P_b is the price of the best bid; and the constants C and M are parameters that determine the ISHV's linear response to the imbalance.

It is important to note that our work differs from the norm in studies of market dynamics because we are primarily focused here on changes in the prices that are quoted by the traders in response to an imbalance occurring at the top of the LOB, *before* the next transaction takes place, rather than on the nature of the longer-term time series of a sequence of successive transactions in that market. Because the system is inherently stochastic, we repeat this basic shock-test some number of times, and then generate relevant visualizations and calculate summary statistics.

Figure 4 shows the typical price dynamics of a market populated homogeneously by 20 ISHV traders (i.e., $N=10$) and in which no quantity shock occurs. Fresh

customer orders (i.e, exogenous commands to either buy or to sell a specified quantity of the asset, with an associated maximum-purchase-price or minimum-sale-price) are assigned to the 10 buyers and the 10 sellers every 20 seconds, starting at $t=20$: the graph shows the prices quoted by each of the 10 buyers and 10 sellers. For each ISHV trader, after it has been assigned a customer order, it then quotes a price when prompted to: if its side of the LOB is empty, the price it quotes is based on the limit price for its current assignment; but if its side of the LOB does already show a best price, then ISHV attempts to improve on that price by calculating its own value for Δs according to the method shown in Figure 3, and then adding that Δs to the best price (when the assignment is to buy) or subtracting it (when the assignment is to sell). As can be seen from Figure 4, the prices quoted by a population of ISHV buyers and sellers converge and meet at a value close to (but not necessarily exactly equal to) P_0 , and transactions then occur when the prices quoted by the traders start to cross the spread. The convergence is steady and stable (i.e., once converged, the transaction prices remain close to P_0), and repeats once every 20 seconds as fresh assignments are distributed to the traders. Figure 5 shows the market impact effects of a single buy order suddenly increasing the quantity supplied, pushing the prices up rapidly *before* a transaction occurs; for brevity we do not show here a similar figure illustrating the corresponding market impact effect of a single sell order suddenly increasing the quantity demanded, but the dynamics in that case are very similar: arrival of a sell block order pushes the prices down *before* a transaction occurs. In both cases, arrival of a block order triggers a price move against the block trader: when trying to execute a large buy order, quoted prices rise; when trying to execute a large sell order, quoted prices fall; and in both cases the price shift occurs before any transaction takes place. These results demonstrate that, in the single lit-pool version of BSE populated by ISHV traders, market impact is as genuine a concern as in real markets: that is, the dynamics of our model market do capture size impact effects, and so can function as a valuable first approximation for studying impact effects in real-world markets. Having established that our simulated market traders can now demonstrate market impact effects, the next section briefly describes our extension of BSE to incorporate a coupled dark pool, and discusses its effectiveness in reducing market impact.

4. DARK POOL CUTS MARKET IMPACT

We altered the public-domain source-code of BSE, extending its functionality to include a coupled pair of dark and lit pools. In the first instance, the two pools are identical objects: each has the full LOB functionality of the original BSE lit-only trading pool, except that that details of orders sat on the dark pool are not published by the exchange, i.e. the traders do not know any details of orders sat on the dark LOB, other than their own orders. A very simple order-routing mechanism was implemented whereby, when the exchange receives an order, if the size of the order is below a threshold value then the order is "small" and is sent to the lit pool for processing, whereas orders larger than the threshold are "block" sized and hence processed by the dark pool.

Because ISHV traders use only the lit-pool microprice, this simple mechanism is sufficient to eliminate market impact effects for orders sized above the block threshold: now larger orders do not appear on the lit LOB, and so the prices quoted by the population of ISHV traders are now unaffected by the arrival of any such block orders.

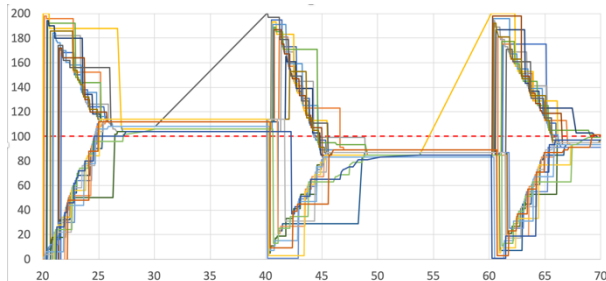


Figure 4 time-series of the prices being quoted by 20 ISHV traders (10 buyers and 10 sellers) across three successive batch assignments of customer orders: horizontal axis is time; vertical axis is price; each trader's current quote-price is represented by a single line. Fresh customer orders, one order per trader and each order for a quantity of one, are redistributed every 20 seconds: price competition among the sellers and buyers results in convergence toward prices close to the P_0 value of \$1.00 (indicated by the horizontal dashed line). After each batch of assignments, the actual price converged to varies slightly from the underlying P_0 value, but the deviation is never large.

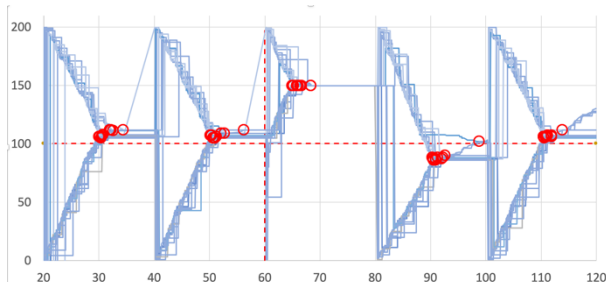


Figure 5: quote-prices and transaction prices showing quantity-impact effect in a market populated entirely by ISHV traders. These data come from the same structure of experiment as described in the caption to Figure 4, but here all the traders' individual quote-prices are the same pale blue colour, and the time and price of individual transactions are illustrated as red circles. At time $t=60$ (indicated by the vertical dashed line), a single large buy order (quantity=200) is assigned to one trader, imposing a step-change quantity shock on the market's demand schedule. The ISHV strategy detects the imbalance between supply and demand at the top of the LOB and reacts by rapidly increasing buyer's quote prices before any transactions take place: transactions then occur at prices around \$1.50, a large deviation from P_0 . After that, the large order is removed from the market and prices return to the previous equilibrium.

It is important to note that, in practice, the effectiveness of such a simple coupled lit/dark pair of pools is potentially open to exploitation by unscrupulous traders who deliberately submit orders with sizes large enough to be routed to the dark pool, but who then delete those orders once they have partially completed (i.e., once they have been matched with one or more smaller orders). To guard against this, LSE Turquoise has mechanisms for tracking the "reputation" of traders who use the dark pool, and only traders with a sufficiently high reputation

are permitted ongoing use. Our simulation of the LSE Turquoise reputation-based access system, described in detail in (Church, 2019), is discussed in the next section.

5. REPUTATIONAL DARK POOL ACCESS

Turquoise Lit is a LOB-based exchange in which the LOB is made visible to all participants, and hence Turquoise Lit can be simulated by the existing BSE functionality as documented by Cliff (2018a, 2018b) and does not need to be discussed further here. However, BSE does not have any dark-pool functionality, nor any reputation-scoring system, and so the focus in this section is on describing those two aspects of LSE Turquoise. In Section 5.1 we introduce the Plato dark pool, and then in Sections 5.2 to 5.6 we introduce various aspects of the Plato operation that are involved in the reputational scoring, which is described in Section 5.7.

5.1. The Turquoise Plato Dark Pool

Plato is the dark pool service offered by Turquoise. Orders submitted to Turquoise Plato are added to the Turquoise Plato Order Book, which we refer to here as TPOB. TPOB is not made visible to the participants in Turquoise Plato, as is expected of a dark pool trading venue; however, the details of each trade that takes place are subsequently made public, after the trade completes. The Turquoise documentation refers to bids as *buy orders* and asks as *sell orders*, so we use that terminology here. Buy orders rest on the buy side of TPOB and sell orders rest on the sell side of TPOB. All trades that take place within Turquoise Plato execute at the midprice of the LOB on the primary (i.e., lit) market for the financial instrument being traded. The primary market is the trading venue where the financial instrument was first admitted to trading. Orders submitted to Turquoise Plato can specify an optional limit price. For an order with a limit price to be able execute, the limit price must be better than the current primary market midprice. Orders can also specify an optional Minimum Execution Size (MES). The MES specifies the minimum quantity that can be traded in that order. For example, say that a trader submits a buy order X with a quantity of 20 and a MES of 10. Say that another trader submits a sell order Y with a quantity of 15 and a MES of 12. These orders can be matched for a trade since the quantity of each order is greater than the MES of the other order. A quantity of 15 will be traded in this case. Now, let's say that instead of sell order Y, a sell order Z was submitted with a quantity of 9 and a MES of 5. In this case the buy order X and sell order Z will not be matched for a trade since the quantity of Z is less than the MES of X. Orders also have a duration attribute which determines how long the order will rest on TPOB, if at all. The duration can be set as *Fill-or-Kill* (FOK), which means that the order must be able to execute immediately and in full or it will not be executed at all, and will not be added to TPOB. The duration can also be set as *Immediate-or-Cancel* (IOC), meaning that the order must be able to execute immediately either fully or partially otherwise it will be cancelled; if the order is only partially executed then the remaining quantity of the order will not be added to TPOB. The duration of an order may also be specified as an expiration time which will cause the order to rest on TPOB until either it is fully executed or the expiration

time is reached. Orders that lie on TPOB until execution or expiry are known as *persistent* orders.

Persistent orders with larger quantities are given a higher priority when order-matching occurs. If two orders have the same quantity, then the order with the earlier arrival time is given the higher priority. If an order is only partially executed, the remaining unexecuted portion of that order maintains the priority that the whole order was first assigned on the basis of its initial quantity.

When trying to match persistent orders, matching starts with the highest-priority buy order. The sell orders are then checked in priority order, from highest to lowest, to see if they can match with the buy order for a trade. If no match is found for this buy order, then the next highest priority buy order is selected for matching, and the process is repeated. When a match is found between a buy order and a sell order, the trade is executed. This process of matching orders and executing trades operates continuously and is referred to as *continuous matching mode*.

5.2. Block Discovery

Turquoise Plato offers an additional service called Block Discovery which is intended to be used by traders that wish to place block orders. Block Discovery allows a trader to first tentatively identify if there are any counterparties that can match their block order before they commit to submitting a firm order. This is done by the trader first submitting a Block Indication (BI). A BI is an indication to the exchange that a trader wants to place a block order, but this is not yet a firm order. Within the BI, the trader specifies the details of the firm order that they are interested in submitting. This includes the quantity, the limit price, the MES, and the duration. Turquoise's matching logic then attempts to match the trader's BI with another BI submitted by a different trader. Once a match between two BIs is found, the traders that submitted the BIs are each sent an Order Submission Request (OSR) notifying them of the match. The OSRs ask the traders to convert their BIs into firm orders by sending a Qualifying Block Order (QBO). The QBO specifies the final details of the firm order that the trader will now be submitting. The details that a trader specifies in a QBO can be different from the original details specified in the BI. Each trader has a reputational score, and if the details specified in the submitted QBO are different from the corresponding BI, then the trader's reputational score may be negatively affected. Once the QBOs are received from both traders, the firm orders specified in the QBOs are submitted to the TPOB. The matching of these orders and the execution of the trade will then take place. To summarize, the main steps that occur in the Block Discovery process are: (1) a trader submits a BI; (2) if a match is found for that BI, then the trader is sent an OSR to notify them; (3) the trader places a firm order by submitting a QBO; (4) the firm order is added to the TPOB and the trade takes place; (5) the trader's reputational score is updated.

5.3. Block Indication (BI)

A BI is not a firm order, but an indication to the Block Discovery service that the trader is interested in placing a firm order. The trader first wants to see if any counterparty can match their order. The BI contains

details about the firm order that the trader would like to place, such as the quantity, the limit price, the MES, and the duration. The limit price and the MES are optional and can be omitted. The quantity specified in the BI must be greater than the Minimum Indication Value (MIV) for the financial instrument being traded. If the quantity is not greater than the MIV, then the BI will be rejected by the Block Discovery service. Whenever a BI is added to the exchange, a check is performed to see if any submitted BIs can be matched for trading. The priority of each BI in the matching process is determined in the same way as it is for orders. If the BI specifies an MES and/or a limit price, then these values are considered in the matching process. Once a match is found for a BI, then the trader is sent an OSR to notify them.

5.4. Order Submission Request (OSR)

An Order Submission Request is sent to a trader to notify them when a match is found for a BI that they submitted. The OSR does not include any details about the counterparty or the counterparty's BI that it was matched with. The OSR is also used to notify the trader of their reputational score. Upon receiving an OSR, a trader is expected to respond with a QBO in order to submit a firm order.

5.5. Qualifying Block Order (QBO)

A Qualifying Block Order is sent by a trader to the exchange upon receipt of an OSR. The OSR tells the trader that their BI has been matched with another BI on the exchange. The QBO is a confirmation of their BI; turning their indication into a firm order. The details specified in the QBO may be different from the details specified in the original BI. This recognizes that sometimes a trader's preferences or circumstances change after a BI is submitted. For example, the QBO could specify a smaller quantity or a higher limit price than in the original BI. The differing details between a QBO and the corresponding BI will affect the trader's reputational score. If a QBO's details are sufficiently different from the original BI that the trade becomes impossible, this will have a negative impact on the trader's reputational score. Once the QBOs are received from both traders, the firm orders are added to the TPOB and the trade can take place.

5.6. Block Discovery Notification (BDN)

When a trader submits an order, they can specify whether the order is a Block Discovery Notification (BDN). If an order is a BDN, then the order is eligible for participation in the Block Discovery service. This means that the order may end up getting matched with a BI. An order can only be a BDN if the quantity of the order is greater than the Minimum Notification Value for the financial instrument being traded. If the order is matched with a BI, then the trader that submitted the BI is notified with an OSR as usual. The trader that submitted the BI is not made aware that they have been matched with a BDN order.

5.7. Reputational Scoring

The Block Discovery service monitors each trader's conversion from BIs to QBOs with a reputational scoring system. A trader is given an event reputational score (ERS) every time they convert a BI into a QBO. A QBO should be *marketable* in comparison to its corresponding

BI. In relation to price, a QBO is marketable if it meets any of the following criteria:

- The BI does not specify a limit price and the QBO does not specify a limit price.
- The BI specifies a limit price and the QBO does not specify a limit price.
- The BI specifies a limit price and the QBO specifies a more marketable limit price. When buying, this means that the QBO limit price must be greater than the BI limit price. When selling, this means that the QBO limit price must be less than the BI limit price.

In relation to the MES, a QBO is marketable if it meets any of the following criteria:

- The BI does not specify an MES and the QBO does not specify an MES.
- The BI specifies an MES and the QBO does not specify an MES.
- The BI specifies an MES, and the QBO specifies an MES that is less than or equal to the BI's MES.

If a QBO is not marketable, then the ERS given to the trader is zero. If the QBO is marketable, then the ERS is calculated by considering the difference in the quantity specified in the BI and the quantity specified in the QBO. The resulting ERS will be between 50 and 100. The exact formula used to calculate the score in this case is not explicitly specified in the public documentation for *Turquoise Plato*, so in our simulation we have a heuristic rule that can be edited to explore different approaches.

A trader's composite reputational score (CRS) is calculated from a weighted sum of their last 50 individual ERSs, with more recent scores having higher weights. A trader is notified of their CRS in every OSR that they receive. If a trader's CRS falls below the exchange's Reputational Score Threshold (RST), then the trader will no longer have access to the Block Discovery service. A trader's CRS persists from one day to the next; any trader with a CRS less than the RST can in principle increase it over time by issuing a succession of BIs and matching QBOs where each QBO does meet the commitments of the corresponding BI.

Figure 6, from Church (2019), shows how an individual trader's CRS changes over time when the trader consistently responds with a QBO having the exact same details as the corresponding originally submitted BI. The trader receives an ERS of 100 every time they convert a BI into a QBO which results in the trader's CRS rising over time.

Figure 7, also from Church (2019), shows how an individual trader's CRS changes over time when it consistently returns a QBO with a quantity half that of the originally submitted BI. This results in the trader receiving an ERS of 50 every time they convert a BI into a QBO. This causes the trader's CRS to fall from its initial value over time. Eventually, the trader's CRS falls below the RST: at this point, the trader is no longer allowed to participate in the Block Discovery service.

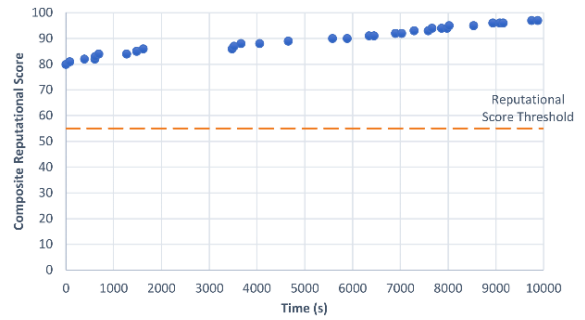


Figure 6: rise in a single trader's composite reputational score over time, when the trader is consistently responding with QBOs that match the corresponding BIs.

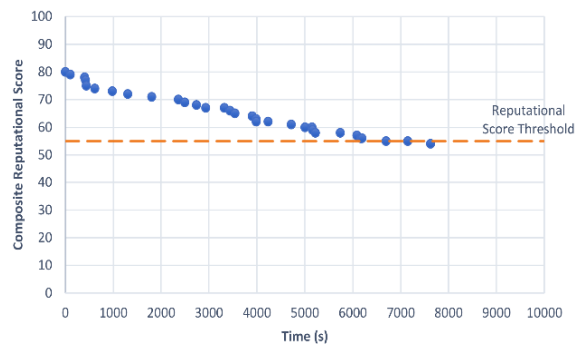


Figure 7: fall in a single trader's composite reputational score over time, when the trader is consistently responding with QBOs that are under-sized with respect to the corresponding BIs: eventually the composite reputational score falls below the system's RST and the trader can no longer access the Block Discovery service.

6. FUTURE WORK

Currently one aspect of our BSELD simulator that requires further work is the implementation of functionality that models *Turquoise Plato Uncross*. Uncross events occur many times in each trading day. These events are triggered by the matching of block indications, or at random time intervals if no triggers have fired for a sufficiently long time.

Further work could also be devoted to taking ISHV's use of the difference between the market's midprice and microprice as an indicator of order-book imbalance likely to have an impact effect, and incorporating that or a similar mechanism into well-known high-performing trading strategies such as AA, GDX, or ZIP.

7. SUMMARY AND CONCLUSIONS

To the best of our knowledge, this is the first published account of a functional simulation of the LSE *Turquoise* coupled lit/dark pools and associated reputation-based order management system. For us to demonstrate the utility of BSELD, it was necessary to create a new type of automated trading agent, one that is sensitive to imbalances in the quantities on the limit order book and which in consequence exhibits market impact behaviors in its pricing: we have done that with ISHV, presented here for the first time, and as far as we are aware ISHV is the only public-domain automated trader that can be used in this way, to explore market impact issues. Thus

the two primary contributions of this paper are the first-ever demonstration of size-related impact effects in markets populated by automated traders, via our introduction of the ISHV strategy; and the reduction (to the point of elimination) of those impact effects via the introduction of mechanisms in BSELD that are closely modelled on the LSE Turquoise trading platform.

By adding the code for our BSELD simulator, and the ISHV trader, to the public-domain BSE repository on GitHub we are providing a tool that can be widely used as the platform for a large range of further research in experimental economics, market microstructure, automated trading, and computational finance. Providing BSELD as a freely-available shared common platform for such research is intended to facilitate replication and extension of results, and to ease the work of researchers wanting to study the behavior of automated trading systems in truly contemporary market environments.

8. REFERENCES

- Arnuk, S., & Saluzzi, J., 2012. *Broken Market: How High-Frequency Trading and Predatory Practices on Wall St are Destroying Investor Confidence and your Portfolio*. Financial Times / Prentice Hall.
- Bodek, H. & Dolgoplov, S., 2015. *The Market Structure Crisis: Electronic Stock Markets, High Frequency Trading, and Dark Pools*. CreateSpace Publishing.
- BSE, 2012. *The Bristol Stock Exchange*. On GitHub at github.com/davecliff/BristolStockExchange
- Cartea, A., Jaimungal, S., & Penalva, J., 2015. *Algorithmic and High-Frequency Trading*. Cambridge University Press.
- Church, G., 2019. *Extending the Bristol Stock Exchange with a Dark Pool*. Master's Thesis, Department of Computer Science, University of Bristol.
- Cliff, D., 1997. *Minimal-Intelligence Agents for Bargaining Behaviours in Market-Based Environments*. Hewlett-Packard Labs Technical Report HPL-97-91.
- Cliff, D., 2018a. BSE: A Minimal Simulation of a Limit-Order-Book Stock Exchange. In: M. Affenzeller, A. Bruzzone, et al. (eds) *Proceedings of the European Modelling and Simulation Symposium (EMSS2018)*, pp.194–203.
- Cliff, D., 2018b. “An Open-Source Limit-Order-Book Exchange for Teaching and Research.” *Proc. IEEE Symposium on Computational Intelligence in Financial Engineering (CIFEr)*, pp.1853–1860.
- Cliff, D., 2019. “Exhaustive Testing of Trader-agents in Realistically Dynamic Continuous Double Auction Markets: AA Does Not Dominate”. In: *Proceedings ICAART 2019*, Vol.2: 224–236; ScitePress.
- Das, R., Hanson, J., Kephart, J., & Tesauro, G., 2001. “Agent-Human Interactions in the Continuous Double Auction”. *Proc. International Joint Conference on AI (IJCAI'01)*, pp.1169–1176.
- De Luca, M., & Cliff, D., 2011. Human-Agent Auction Interactions Adaptive-Aggressive Agents Dominate. *Proceedings of the International Joint Conference on AI (IJCAI-2011)*, pp.178–185.
- De Luca, M., 2015. *Adaptive Algorithmic Trading Systems*. PhD Thesis, Department of Computer Science, University of Bristol, UK.
- Gjerstad, S. & Dickhaut, J., 1998. Price Formation in Double Auctions. *Games & Economic Behavior*, 22(1):1–29.
- Gjerstad, S., 2003. *The Impact of Pace in Double Auction Bargaining*. Working Paper, Department of Economics, University of Arizona.
- Gode, D. & Sunder, S., 1993. Allocative efficiency of markets with zero-intelligence traders: Market as a partial substitute for individual rationality. *Journal of Political Economy*, 101 (1):119–137.
- Harris, L., 2002. *Trading and Exchanges: Market Microstructure for Practitioners*. Oxford University Press.
- London Stock Exchange Group, 2019a. *Turquoise Plato Block Discovery (V2.26.2)*: bit.ly/2ZPJAi0.
- London Stock Exchange Group, 2019b. *Turquoise Trading Service Description (V3.34.9j)*: bit.ly/2Y1jWFi.
- Narang, R., 2013. *Inside the Black Box: The Simple Truth about Quantitative Trading*. Second Edition. Wiley Finance.
- Palmer R., Rust, J., & Miller, J. 1992. Behavior of Trading Automata in a Computerized Double Auction Market. In D. Friedman, & J. Rust (Eds.), *Double Auction Markets: Theory, Institutions, Laboratory Evidence*. Addison Wesley.
- Patterson, S., 2013. *Dark Pools: The Rise of AI Trading Machines*. Random House.
- Pentapalli, M., 2008. *A comparative study of Roth-Erev and Modified Roth-Erev reinforcement learning algorithms for uniform-price double auctions*. PhD Thesis, Iowa State University.
- Rodgers, K. 2016. *Why Aren't They Shouting? A Banker's Tale of Change, Computers, and Perpetual Crisis*. RH Business Books.
- Stotter, S., Cartlidge, J., and Cliff, D. 2013. “Exploring assignment-adaptive (ASAD) trading agents in financial market experiments”, *Proceedings ICAART2013*, 1:77–88.
- Tesauro, G. and Das, R. 2001. “High-performance Bidding Agents for the Continuous Double Auction”. *Proceedings of the 3rd ACM Conference on Electronic Commerce*, pp.206–209.
- Tesauro, G. and Bredin, J., 2002. “Sequential Strategic Bidding in Auctions using Dynamic Programming”. In *Proceedings AAMAS 2002*.
- Vach, D., 2015. *Comparison of Double Auction Bidding Strategies for Automated Trading Agents*. Master's Thesis, Charles University in Prague.
- Vytelingum, P., 2006. *The Structure and Behaviour of the Continuous Double Auction*. PhD Thesis, University of Southampton.
- Vytelingum, P., Cliff, D., & Jennings, N., 2008. “Strategic Bidding in Continuous Double Auctions”. *Artificial Intelligence*, 172(14):1700–1729.